Indicator Variables

Math

- If the same task is performed for different values of a variable, use an indicator variable and vectorize.
- If different tasks are performed for different values of a variable, use a `switch` conditional.
- If different tasks are performed under different conditions, use an `if` conditional.
Different tasks are performed for $x = v_1$, for $x = v_2$ or $v_3$, and for every other value of $x$.

```java
switch x
    case v1
        ...
    case \{v2, v3\}
        ...
    otherwise
        ...
end
```
Different tasks are performed if $x \neq 0$, if $x = 0$ but $y \neq 0$, and if $x = 0$ and $y = 0$.

```
1 if  x
2     ...
3 elseif y
4     ...
5 else
6     ...
7 end
```
if $x$ and $if \ x \sim= 0$ represent the same condition. The expression $x \sim= 0$ should be treated as a variable whose value is
\[
\begin{cases}
1 & \text{if } x \neq 0 \\
0 & \text{if } x = 0
\end{cases}
\]

while $x$ and $while \ x \sim= 0$ represent the same loop for the same reason.
Conditionals Quiz Questions

Quiz
Number of Input Arguments

Code

- When the function `function z = f(x, y)` is called, 0, 1 or 2 arguments can be provided.
- `switch` can be used here to perform different tasks when different number of arguments are given.
- `nargin` is the number of input arguments provided when the function is called.
Log with Optional Input Arguments

Code

For example, a new log function can be defined by \( \log() \) returns 1, \( \log(x) \) returns natural \( \log(x) \), and \( \log(n, x) \) returns \( \log_n(x) \).

```matlab
function z = log(x, y)
    switch nargin
    case 1
        z = log(x);
    case 2
        z = log(y) / log(x);
    otherwise
        z = 1;
    end; end
```
Variable Length Input Argument

Code

- \textit{varargin} represents an arbitrary number of input variables.
- It can only be used as the last argument of a function, for example, \textit{function } \textit{y = f(x1, x2, x3, varagin)}.
- The \textit{i}-th argument can be accessed by \textit{varargin\{i\}}.
For example, a new log function can be defined so that it returns a vector if more than one input is provided.

```matlab
function z = log(x, varargin)
    if nargin == 1
        z = log(x);
    else
        z = [log(x) zeros(1, nargin - 1)];
        for t = 2:nargin
            z(t) = log(varargin{t - 1});
        end; end; end
```
Output Arguments

Code

- `varargout` represents an arbitrary number of output variables.
- `nargout` represents the number of output variables assigned when the function is called.
- For example, $x = \text{size}([1 \ 2; \ 3 \ 4])$ assigns $x$ the value $2 \ 2$ and $[x, \ y] = \text{size}([1 \ 2; \ 3 \ 4])$ assigns $x$ the value $2$.
A function that uses itself in the body is called a recursive function.

1. \textit{function } \ z = f(x) \\
2. \textit{if } \ x \ldots \% \text { base case} \\
3. \quad z = \ldots \\
4. \textit{else } \% \text { recursion} \\
5. \quad z = \ldots \ f(x') \ldots \\
6. \textit{end}
To compute the factorial of $n \geq 0$:

```
1  function z = f(x)
2     if ~x
3         z = 1;
4     else
5         z = x * f(x - 1);
6     end
7  end
```
Recursion Example, Vector Sum

Code

To compute the sum of the values in a vector $v$:

```matlab
function z = f(x, t)
    if nargin == 1
        z = f(x, 0);
    elseif t > length(x)
        z = 0;
    else
        z = x(t) + f(x, t + 1);
    end
end
```
Recursion Quiz Questions

Quiz
Blank Slide